Comparison of CoAP Security Protocols

draft-ietf-iotops-security-protocol-comparison-04
Changes between -02 and -04 (Addressing Early IoTDir review by Russ Housley)

— Added text explaining the difference between AKEs and protocols for protection of application data.
— Added a paragraph to introduce the section on underlying layers. Moved "EDHOC Over CoAP and OSCORE" subsection to appendix.
— Security considerations linking to the security considerations for the protocols as well as newer recommendations and best practices.
— Explained key and certificate identifiers. References for the algorithms.
— Added reference to RFC 7250, RFC 9547, and "Performance Comparison of EDHOC and DTLS 1.3 in Internet-of-Things Environments".
— Editorial changes including adding a change log and updated to cTLS-09, which seems relatively stable.
Changes between -02 and -04

2. Underlying layers

[I-D.ietf-core-oscore-groupcomm]. The described overheads are independent of the underlying transport.

The protocols are analyzed with different algorithms and options. The DTLS and TLS record layers are analyzed with and without 6LoWPAN-GHC compression [RFC7400]. DTLS is analyzed with and without Connection ID [RFC9146]. Readers are expected to be familiar with some of the terms described in RFC 7925 [RFC7925], such as Integrity Check Value (ICV). Section 3 compares the overhead of mutually authenticated key exchange, while Section 4 covers the overhead for protection of application data.

2. Underlying Layers

The described overheads in Section 3 and Section 4 are independent of the underlying layers as they do not consider DTLS handshake message fragmentation, how to compose DTLS handshake messages into records, and how the underlying layers influence the choice of application plaintext sizes. The complete overhead for all layers depends on the combination of layers as well as assumptions regarding the devices and applications and is out of scope of the document. This section gives a short overview of the overheads of UDP, TCP, and CoAP to give the reader a high-level overview.

[I-D.ietf-core-oscore-groupcomm]. An AKE and a protocol for the protection of application data serve distinct purposes. An AKE is responsible for establishing secure communication channels between parties and negotiating cryptographic keys used for authenticated encryption. AKE protocols typically involve a series of messages exchanged between communicating parties to authenticate each other's identities and derive shared secret keys. TLS, DTLS, and cTLS handshakes as well as EDHOC are examples of AKEs. Protocols for protection of application data are responsible for encrypting and authenticating application-layer data to ensure its confidentiality, integrity, and replay protection during transmission. The TLS and DTLS record layers, OSCORE, and Group OSCORE are examples of protocols for protection of application data. Section 3 compares the overhead of mutually authenticated key exchange protocols, while Section 4 covers the overhead of protocols for protection of application data. The protocols are analyzed with different algorithms and options. The DTLS and TLS record layers are analyzed with and without 6LoWPAN-GHC compression [RFC7400]. DTLS is analyzed with and without Connection ID [RFC9146]. Readers are expected to be familiar with some of the terms described in RFC 7925 [RFC7925], such as Integrity Check Value (ICV).
5. Security Considerations

This document is purely informational.

Changes between -02 and -04

5. Security Considerations

When using the security protocols outlined in this document, it is important to adhere to the latest requirements and recommendations for respective protocol. It is also crucial to utilize supported versions of libraries that continue to receive security updates in response to identified vulnerabilities.

While the security considerations provided in DTLS 1.2 [RFC6347], DTLS 1.3 [RFC9147], TLS 1.2 [RFC5246], TLS 1.3 [RFC8446], cTLS [I-D.ietf-tls-_tls], EDHOC [I-D.ietf-ake-ehoc], IETF-CORE [I-D.ietf-core-socket-ehoc], OSCORE [RFC8613], Group OSCORE [I-D.ietf-core-socket-groupcomm], and X.509 [RFC5280] serve as a good starting point, they are not sufficient due to the fact that some of these specifications were authored many years ago. For instance, being compliant to the TLS 1.2 [RFC5246] specification is considered very poor security practice, given that the mandatory-to-implement cipher suite TLS_RSA_WITH_AES_128_CBC_SHA possesses at least three major weaknesses.

Therefore, implementations and configurations must also align with the latest recommendations and best practices. Notable examples when this document was published include BCP 195 [RFC9325][RFC9996], [SP-800-52], and [BSI-TLS].
Next steps

— Two informative references in AUTH48
  — RFC 9528 (EDHOCS) and RFC 9529 (EDHOCS Traces)
— Editor's note: This version of the document analyses the -09 version of cTLS, which seems relatively stable. It is uncertain if the TLS WG will adopt more compact encoding for P-256 and ECDSA such as secp256r1_compact and ecdsa_secp256r1_sha256_compact {{I-D.mattsson-tls-compact-ecc}}.
— All other issues and comments have been addressed.
— More reviews, WGLC?